

Error bounds on the probabilistically optimal problem solving strategy

Frantisek Duris*

Department of Computer Science,
Faculty of Mathematics, Physics and Informatics,
Comenius University,
Mlynska dolina 842 48 Bratislava

December 1, 2016

Abstract

We consider a simple optimal probabilistic problem solving strategy that searches through potential solution candidates in a specific order. We are interested in what impact has interchanging the order of two solution candidates with respect to this optimal strategy on the problem solving time. Such interchange (i.e., error) can happen in the applications with only partial information available. We derive bounds on the excessive problem solving time due to these errors in general case as well as in some special cases in which we impose restrictions on the solution candidates.

Keywords: optimal problem solving strategy, error bounds, effectivity, artificial intelligence

MSC number: 68T20 Problem solving

*fduris@dcs.fmph.uniba.sk

1 Introduction and layout of the paper

Problem solving is probably the most common mental activity of humans, and if artificial intelligence is to succeed in creating human like agents, it must deal with the design of general problem solving algorithm. One reason why the human like agents are desirable in the first place is the effectivity with which the human solvers manage to solve vast amount of problems. Moreover, the same human brain (i.e., the same underlying mechanisms) can solve problems from a very wide variety (mathematics, logic, pattern recognition, orientation in space, coordination to name just a few). Solomonoff [13] in his work on general system for solving problems described a basic model of problem solving process.

There is a casino with a set of bets each winning the same prize. The k^{th} bet has probability of winning p_k and costs d_k dollars. All probabilities are independent, and one cannot make the same bet twice. The probabilities p_k do not need to sum to 1 (i.e., there can be three bets with 50% chance of winning). If all bets cost equally (e.g. one dollar), then, obviously, the best strategy is to take the bet with highest win probability available. If not all d_k are same, then by selecting the bet with highest ratio p_k/d_k available the expected money spent before winning will be minimal (proof given below). By changing dollars d_k to time t_k we get the least expected time before winning.

The problem solving interpretation of this scenario is straightforward, and it outlines the optimal probabilistic problem solving strategy provided we know the values p_k and t_k for all solution candidates (i.e., probability of solving the problem and time of execution for each solution candidate). Admittedly, this requirement of knowing p_k and t_k for all solution candidates is very strong and generally hard to meet. However, in our paper *Theory of the effectivity of human problem solving* submitted for publication we argued that humans have access to these values or at least to their approximations and that they use them (consciously or unconsciously) to drive their problem solving according to the Solomonoff optimal strategy. It is for this reason we are interested in the Solomonoff optimal problem solving strategy. Moreover, since humans do not always follow the Solomonoff optimal problem solving strategy (inaccurate approximations of p_k or t_k , or other reasons), we are interested in what effect the interchange of two candidates has on the problem solving time.

The importance of automatic selection of the problem solving methods the has long been noted [7, 8, 2]. For example, Fink [1] developed a statistical method for automatic selection among methods based on analysis of their past performance. More particularly, Fink analyzed the expected gain (there is a reward for solving the problem) of each available method based on its past success', failures and the corresponding execution times. The proposed system for solving problems not only chooses the most efficient method (given its limited available data and a time bound), but also decides when to abandon the chosen method if it takes too much time.

Solomonoff' in his work on the general system for solving problems used his theory of inductive inference [10, 11] to approximate the probabilities p_k and Levin search [6, 12] to approximately follow the optimal order of examination of candidates as outlined in his betting

scenario [13]. Since this is an approximation of the optimal order, it does not need to follow it exactly, and we might be interested in what effect the shuffling of candidates has on the problem solving time. Other problem solving systems that are also related to the Solomonoff problem solving strategy and the approximation of probabilities p_k (e.g., AIXI proposed by Hutter [3, 4]) may benefit from this paper in the similar way.

Schmidhuber [9] described a similar notion of time optimality of the problem solving. While he also uses the values p_k and t_k , he does not try to approximate the probabilities p_k (termed initial bias) as Solomonoff. Instead, he assumes that they are given as input (although they may change during the problem solving process; the same applies to the Solomonoff problem solving system). Again, the changes in the initial probabilities may lead to different order of candidate examination, and it is interesting to ask what effect does this have on the problem solving time.

The layout of the paper is as follows. In *Section 2* we prove the optimality of the strategy mentioned in the introduction as Solomonoff [13] did not do so. The proof will serve as a basis for all our subsequent results. In *Section 3* we consider the effect of interchanging two candidates with respect to the optimal strategy on the problem solving time. In *Section 4* we give several bounds on the error resulting from the mentioned interchange. However, since the values p_k and t_k can be arbitrary, we examine special cases under which reasonable bounds can be achieved. In *Section 5* we consider a modification of this strategy when each solution candidate can have multiple values of execution time (i.e., multiple values t_k). This modification models the case when we applied the same solution candidate (e.g., a method) to two or more similar problems each time solving the problem in different time.

2 Theorem in probability

Let there be a set of bets each winning the same prize. Let p_k denote a probability of winning with k^{th} bet, and let d_k denote the cost of this bet. Each bet can only be taken once, and the values p_k do not need to sum to 1. One strategy to win is to take bets in the order given by decreasing value $\frac{p_k}{d_k}$. This strategy appeared in Solomonoff paper on general problem solving system [13], but the proof was omitted. In this section we give proof of this statement because it will serve as a basis for all of our subsequent results. Additionally, since we would like to apply this strategy in real world, we restrict the number of bets to some finite number N .

Theorem 2.1 (Solomonoff [13]). *Let there be a set of N bets $\{s_k\}_{k=1}^N$, each with probability of winning p_k and cost d_k . If one continues to select subsequent bets on the basis of maximum p_k/d_k , the expected money spent before winning will be minimal. Suppose we change dollars to some measure of time t_k . Then, betting according to this strategy yields the minimum expected time to win.*

Remark 2.2. Note that if the bets are selected in the order: $1^{st}, 2^{nd}, \dots, N^{th}$, then the probability of using and winning with a particular bet k is not p_k but $\prod_{i=1}^{k-1} (1 - p_i) \cdot p_k$. This is because in order to make and win with the k^{th} bet all bets with the indices $1, 2, \dots, k - 1$ must

have failed.

Proof of the Theorem 2.1. Without loss of generality, we may assume that the sequence of bets $(s_k)_{k=1}^N$ is ordered according to the values $\frac{p_k}{t_k}$ in the decreasing order, i.e.,

$$\frac{p_1}{t_1} \geq \frac{p_2}{t_2} \geq \dots \geq \frac{p_k}{t_k} \geq \dots \geq \frac{p_N}{t_N}, \quad (2.1)$$

in which case the Solomonoff strategy SOL is $(s_k)_{k=1}^N$. Let E_T be the expected time spent before winning using the Solomonoff strategy SOL . Clearly,

$$E_T = \sum_{k=1}^N \sum_{l=1}^k t_l \cdot \prod_{j=1}^{k-1} (1 - p_j) \cdot p_k.$$

We want to show that this strategy is optimal (with respect to the time spent before winning). Let $ABC = (s_{i_k})_{k=1}^N$ be any betting strategy (i.e., a sequence of bets). Furthermore, let E_{ABC} be the expected time spent before winning for the strategy ABC . Clearly,

$$E_{ABC} = \sum_{k=1}^N \sum_{l=1}^k t_{i_l} \cdot \prod_{j=1}^{k-1} (1 - p_{i_j}) \cdot p_{i_k}.$$

Our aim is to show that $E_{ABC} \geq E_T$. If $ABC = SOL$, then we have nothing to prove. Now assume that there are two immediately subsequent bets s_{i_a} and $s_{i_{a+1}}$ in the sequence ABC such that $\frac{p_{i_a}}{t_{i_a}} < \frac{p_{i_{a+1}}}{t_{i_{a+1}}}$. The case when $\frac{p_{i_a}}{t_{i_a}} \geq \frac{p_{i_{a+1}}}{t_{i_{a+1}}}$ for each $a \in \{1, 2, \dots, N-1\}$ but $ABC \neq SOL$ will be considered below. Let ABC' be a modified sequence ABC in which the terms s_{i_a} and $s_{i_{a+1}}$ are interchanged. We will show that $E_{ABC} \geq E_{ABC'}$ where $E_{ABC'}$ denotes the analogous value for ABC' . First of all, notice that all terms in E_{ABC} and $E_{ABC'}$ are equal except for the terms on the a^{th} and $(a+1)^{th}$ position. In the expression E_{ABC} we have the following value related to these two positions

$$\sum_{l=1}^a t_{i_l} \cdot \prod_{j=1}^{a-1} (1 - p_{i_j}) \cdot p_{i_a} + \sum_{l=1}^{a+1} t_{i_l} \cdot \prod_{j=1}^a (1 - p_{i_j}) \cdot p_{i_{a+1}},$$

while in the expression $E_{ABC'}$ there is

$$\left(\sum_{l=1}^{a-1} t_{i_l} + t_{i_{a+1}} \right) \cdot \prod_{j=1}^{a-1} (1 - p_{i_j}) \cdot p_{i_{a+1}} + \left(\sum_{l=1}^{a-1} t_{i_l} + t_{i_{a+1}} + t_{i_a} \right) \cdot \prod_{j=1}^{a-1} (1 - p_{i_j}) \cdot (1 - p_{i_{a+1}}) p_{i_a}.$$

Therefore,

$$\begin{aligned} E_{ABC} - E_{ABC'} &= \left(\sum_{l=1}^{a-1} t_{i_l} + t_{i_a} \right) \cdot \prod_{j=1}^{a-1} (1 - p_{i_j}) \cdot p_{i_a} - \\ &\quad - \left(\sum_{l=1}^{a-1} t_{i_l} + t_{i_{a+1}} \right) \cdot \prod_{j=1}^{a-1} (1 - p_{i_j}) \cdot p_{i_{a+1}} + \\ &\quad + \left(\sum_{l=1}^{a-1} t_{i_l} + t_{i_a} + t_{i_{a+1}} \right) \cdot \prod_{j=1}^{a-1} (1 - p_{i_j}) (1 - p_{i_a}) p_{i_{a+1}} - \\ &\quad - \left(\sum_{l=1}^{a-1} t_{i_l} + t_{i_{a+1}} + t_{i_a} \right) \cdot \prod_{j=1}^{a-1} (1 - p_{i_j}) (1 - p_{i_{a+1}}) p_{i_a}. \end{aligned}$$

By substituting

$$\sum = \sum_{l=1}^{a-1} t_{i_l} \quad \text{and} \quad \prod = \prod_{j=1}^{a-1} (1 - p_{i_j})$$

and some rearranging we get

$$\begin{aligned} E_{ABC} - E_{ABC'} &= \sum \cdot \prod \cdot (p_{i_a} - p_{i_{a+1}}) + \\ &+ \prod \cdot (t_{i_a} p_{i_a} - t_{i_{a+1}} p_{i_{a+1}}) + \\ &+ \sum \cdot \prod \cdot (p_{i_{a+1}} - p_{i_a}) + \\ &+ \prod \cdot (t_{i_a} + t_{i_{a+1}}) (p_{i_{a+1}} - p_{i_a}) = \\ &= \prod \cdot (t_{i_a} p_{i_{a+1}} - t_{i_{a+1}} p_{i_a}) = \\ &= \prod_{j=1}^{a-1} (1 - p_{i_j}) \cdot t_{i_a} t_{i_{a+1}} \left(\frac{p_{i_{a+1}}}{t_{i_{a+1}}} - \frac{p_{i_a}}{t_{i_a}} \right) \geq 0. \end{aligned}$$

It follows that ABC can be turned into a betting strategy $DEF = (s_{m_k})_{k=1}^N$ with the property $\frac{p_{m_1}}{t_{m_1}} \geq \frac{p_{m_2}}{t_{m_2}} \geq \dots \geq \frac{p_{m_N}}{t_{m_N}}$ by repeatedly modifying the obtained sequences. Moreover, the expected amount of dollars paid over a sequence (i.e., strategy) before winning is not increased after its modification. Thus, $E_{ABC} \geq E_{ABC'} \geq \dots \geq E_{DEF}$. If $DEF \neq SOL$, we proceed as follows. Let DEF' be a modified sequence DEF in which any two terms s_{m_b} and $s_{m_{b+1}}$ with $\frac{p_{m_b}}{t_{m_b}} = \frac{p_{m_{b+1}}}{t_{m_{b+1}}}$ are interchanged (call this kind of modification by simple modification). Using the same calculation as for $E_{ABC} - E_{ABC'}$ above, we get $E_{DEF} - E_{DEF'} = 0$, where $E_{DEF'}$ denotes analogous value for DEF' , since $\frac{p_{m_b}}{t_{m_b}} - \frac{p_{m_{b+1}}}{t_{m_{b+1}}} = 0$. Now one can observe that DEF can be turned into SOL by repeatedly modifying the obtained sequences (using only the simple modification). Thus, $E_{DEF} = E_{DEF'} = \dots = E_{SOL}$.

Finally, let us consider the case when $\frac{p_{i_a}}{t_{i_a}} = \frac{p_{i_{a+1}}}{t_{i_{a+1}}}$ for each $a \in \{1, 2, \dots, N-1\}$, but $ABC \neq SOL$. In such case, we can turn ABC into SOL by the same way as we have turned DEF into SOL above, and therefore $E_{ABC} = E_T$. Consequently, $E_{ABC} \geq E_{DEF} = E_T$ (see above), or $E_{ABC} = E_T$. Hence, SOL is optimal, since the strategy ABC has been chosen arbitrarily. \square

Remark 2.3. Notice that the expected betting time is not given by E_T because we did not include the possibility that all of our bets failed. The corrected value E_T is given by

$$E_T = \sum_{k=1}^N \sum_{l=1}^k t_l \cdot \prod_{j=1}^{k-1} (1 - p_j) \cdot p_k + \sum_{l=1}^N t_l \cdot \prod_{j=1}^N (1 - p_j). \quad (2.2)$$

This is because the probability of each bet failing is $\prod_{j=1}^N (1 - p_j)$ while it takes us altogether $\sum_{l=1}^N t_l$ amount time to discover this.

3 General effects of exchanging the order of two candidates on the problem solving effectivity

In this section we abandon the Solomonoff' original betting scenario, and we will talk about solution candidates (i.e., bets) and problem solving strategy (i.e., betting on candidates). We

consider and quantify the expected decrease in problem solving effectivity when the solver makes an error and interchanges the order of two (not necessarily immediately subsequent) candidates with respect to the optimal Solomonoff problem solving strategy.

Lemma 3.1 (Klamkin and Newman [5]). *If x_1, x_2, \dots, x_n are numbers in $[0, 1]$ whose sum is denoted by S , then*

$$\prod_{i=1}^n (1 - x_i) < e^{-S}.$$

Lemma 3.2 (Wu [14]). *Let $0 \leq x_i \leq 1, i = 1, 2, \dots, n, n \geq 2, n \in \mathbb{N}$. Then we have*

$$\prod_{i=1}^n (1 - x_i) \geq 1 - \sum_{i=1}^n x_i + (n - 1) \left(\prod_{i=1}^n x_i \right)^{\frac{n}{2n-2}}.$$

Remark 3.3. For the rest of this paper we will use the following notation

$$\begin{aligned} S_m &= \sum_{i=1}^m p_i, & T_m &= \sum_{i=1}^m t_i, \\ P_m &= \prod_{i=1}^m p_i, & Q_m &= \prod_{i=1}^m (1 - p_i). \end{aligned}$$

Theorem 3.4. *Let $\frac{p_k}{t_k} - \frac{p_{k+1}}{t_{k+1}} > 0$ for some k (assuming $(\frac{p_i}{t_i})_{i=1}^N$ to be ordered as before in Theorem 2.1). Then, following the optimal Solomonoff strategy from Theorem 2.1 with $(k+1)^{th}$ solution candidate tried just before k^{th} (a solver's error) yields a sub-optimal expected amount of time spent before either finding a solution or discovering that none of our solution candidates works, and the expected excess EXC can be quantified as follows*

$$EXC = \left(\frac{p_k}{t_k} - \frac{p_{k+1}}{t_{k+1}} \right) \prod_{j=1}^{k-1} (1 - p_j) \cdot t_k t_{k+1}.$$

Furthermore,

$$\left(\frac{p_k}{t_k} - \frac{p_{k+1}}{t_{k+1}} \right) t_k t_{k+1} \cdot e^{-S_{k-1}} \geq EXC \geq \left(\frac{p_k}{t_k} - \frac{p_{k+1}}{t_{k+1}} \right) t_k t_{k+1} \cdot \left(1 - S_{k-1} + (k-2)P_{k-1}^{\frac{k-1}{2k-4}} \right).$$

Proof. The theorem follows directly from proof of the Theorem 2.1 (see the expression $E_{ABC} - E_{ABC'}$ and below). To derive the bounds, use Lemma 3.1 and Lemma 3.2. For notation see Remark 3.3. \square

Theorem 3.5. *If $(\frac{p_i}{t_i})_{i=1}^N$ are ordered as before in Theorem 2.1, then exchanging the k^{th} and $(k+n)^{th}$ solution candidates in the optimal Solomonoff strategy from Theorem 2.1 (a solver's error) increases the expected amount of problem solving time by at most the excess*

$$EXC = q_1 + q_2 + q_3,$$

where

$$\begin{aligned} q_1 &= T_{k-1} \cdot Q_{k-1} \cdot (p_{k+n} - p_k) + Q_{k-1} \cdot (t_{k+n}p_{k+n} - t_k p_k), \\ q_2 &= \sum_{l=k+1}^{k+n-1} Q_{l-1} \cdot p_l \left(T_l \cdot \frac{p_k - p_{k+n}}{1 - p_k} + (t_{k+n} - t_k) \frac{1 - p_{k+n}}{1 - p_k} \right), \\ q_3 &= T_{k+n} \cdot Q_{k+n-1} \cdot \frac{p_k - p_{k+n}}{1 - p_k}. \end{aligned}$$

Proof. Use the equation (2.2) and subtract it from an analogous equation in which the k^{th} and $(k+n)^{th}$ candidates are exchanged. The theorem follows by straightforward computation. \square

Remark 3.6. The practicality of the *Theorem 3.4* is considerably limited by the fact that the values $(p_i)_{i=1}^n$ and $(t_i)_{i=1}^N$ can be (almost) arbitrary numbers (obviously, we want $p_i \in [0, 1]$ and $t_i > 0$). Thus, we could not further simplify the terms q_1, q_2, q_3 . In *Section 4* we make some assumptions about these numbers to obtain reasonable bounds on the term EXC from *Theorem 3.4*. Note that these assumptions directly affect the quality of the obtained bounds and the form of the resulting formulas. On the other hand, they also reduce the generality of the derived bounds.

4 More precise quantification of the effects in some special cases

Lemma 4.1. *Let r be any real number other than 1. Then it holds*

$$\sum_{l=1}^n l r^l = \frac{r}{(1-r)^2} (n r^{n+1} - (n+1) r^n + 1).$$

A very general bound of the EXC term from the *Theorem 3.5* follows. Note that we were not able to derive similarly simple lower bound with the same weak assumptions (see *Theorem 4.3*).

Theorem 4.2. *For each $i = 1, 2, \dots, N$, let $c \leq p_i \leq d$ for some $c, d \in (0, 1)$ and $0 \leq t_i \leq T$ for some T . Then, the term EXC from the *Theorem 3.5* can be upper bounded as follows*

$$EXC \leq T \frac{1 - p_{k+n}}{1 - p_k} \frac{d}{c} (1 - c)^k (A - B(1 - c)^{n-1}),$$

where

$$\begin{aligned} A &= 1 + \frac{1}{c} + k + \frac{1 - p_k}{1 - p_{k+n}} \frac{ck}{1 - c}, \\ B &= \left(1 - \frac{c}{d}\right) (k + n) + \frac{1}{c}. \end{aligned}$$

Proof. Let q_1 , q_2 , and q_3 be expressions from *Theorem 3.5*. From $(1 - p_i) \leq (1 - c)$ for all i we have for q_1

$$\begin{aligned} q_1 &= Q_{k-1}p_{k+n}(T_{k-1} + t_{k+n}) - Q_{k-1}p_k(T_{k-1} + t_k) \leq \\ &\leq Q_{k-1}p_{k+n}(T_{k-1} + t_{k+n}) \leq \\ &\leq (1 - c)^{k-1}dkT. \end{aligned}$$

Finally, we transform q_1 into

$$q_1 \leq T \frac{1 - p_{k+n}}{1 - p_k} \frac{d}{c} (1 - c)^k \cdot \frac{1 - p_k}{1 - p_{k+n}} \frac{ck}{1 - c}. \quad (4.1)$$

For q_2 we have

$$\begin{aligned} q_2 &\leq \sum_{l=k+1}^{k+n-1} Q_{l-1}p_l \left(lT \frac{1 - p_{k+n}}{1 - p_k} + T \frac{1 - p_{k+n}}{1 - p_k} \right) = \\ &= T \frac{1 - p_{k+n}}{1 - p_k} \sum_{l=k+1}^{k+n-1} Q_{l-1}(l+1)p_l \leq \\ &\leq T \frac{1 - p_{k+n}}{1 - p_k} d \sum_{l=k+1}^{k+n-1} (1 - c)^{l-1}(l+1) = \\ &= T \frac{1 - p_{k+n}}{1 - p_k} d(1 - c)^{k-1} \left(\sum_{l=1}^{n-1} (1 - c)^l l + (k+1) \sum_{l=1}^{n-1} (1 - c)^l \right). \end{aligned}$$

By using *Lemma 4.1* we can further write

$$\sum_{l=1}^{n-1} (1 - c)^l l = \frac{1 - c}{c^2} (1 + (c - nc - 1)(1 - c)^{n-1}), \quad (4.2)$$

$$(k+1) \sum_{l=1}^{n-1} (1 - c)^l = (k+1)(1 - c) \frac{1 - (1 - c)^{n-1}}{c}. \quad (4.3)$$

Finally, we have for q_2

$$\begin{aligned} q_2 &\leq T \frac{1 - p_{k+n}}{1 - p_k} \frac{d}{c} (1 - c)^k \left(1 + \frac{1}{c} + k + (1 - c)^{n-1} \left(\frac{c - nc - 1}{c} - k - 1 \right) \right) = \\ &= T \frac{1 - p_{k+n}}{1 - p_k} \frac{d}{c} (1 - c)^k \left(1 + \frac{1}{c} + k - (1 - c)^{n-1} \left(k + n + \frac{1}{c} \right) \right). \end{aligned} \quad (4.4)$$

For q_3 we have

$$q_3 \leq (k+n)T(1 - c)^k(1 - c)^{n-1} \frac{1 - p_{k+n}}{1 - p_k}$$

which we transform into

$$q_3 \leq T \frac{1 - p_{k+n}}{1 - p_k} \frac{d}{c} (1 - c)^k \cdot (1 - c)^{n-1} \frac{c}{d} (k+n). \quad (4.5)$$

By putting (4.1), (4.4), and (4.5) together the theorem follows. \square

Theorem 4.3. For each $i = 1, 2, \dots, N$, let $c \leq p_i \leq d$ for some $c, d \in (0, 1)$ and $0 \leq t \leq t_i \leq T$ for some t, T . Furthermore, let $p_k \geq p_{k+n}$ and $t_k \leq t_{k+n}$, where $p_k, p_{k+n}, t_k, t_{k+n}$ are from Theorem 3.5. Then, the term EXC from the Theorem 3.5 can be lower bounded as follows

$$EXC \geq t \frac{p_k - p_{k+n}}{1 - p_k} \frac{c}{d} (1 - d)^k (A - B(1 - d)^{n-1}),$$

where

$$A = \frac{1}{d} + k + \frac{1 - p_k}{p_k - p_{k+n}} dk \left(\frac{1}{1 - d} - \frac{dt}{cT} \frac{e^{-S_{k-1}}}{(1 - d)^k} \right),$$

$$B = \left(1 - \frac{d}{c} \right) (k + n) + \frac{1 - d}{d}.$$

Proof. Let q_1, q_2 , and q_3 be expressions from Theorem 3.5. We rewrite q_1 as

$$q_1 = Q_{k-1} (T_{k-1} + t_{k+n}) p_{k+n} - Q_{k-1} (T_{k-1} + t_k) p_k.$$

By Lemma 3.1 we have $\prod(1 - p_i) \leq e^{-\sum p_i}$ and with $\prod(1 - p_i) \geq \prod(1 - d)$ we can write

$$q_1 \geq (1 - d)^{k+1} ktc - e^{-S_{k-1}} kTd$$

which we transform into

$$t \frac{p_k - p_{k+n}}{1 - p_k} \frac{c}{d} (1 - d)^k \left(\frac{1 - p_k}{p_k - p_{k+n}} \frac{dk}{1 - d} - \frac{1 - p_k}{p_k - p_{k+n}} \frac{d^2 kt}{cT} \frac{e^{-S_{k-1}}}{(1 - d)^k} \right)$$

The rest of the proof is analogous to the proof of the Theorem 4.2. \square

A different bound of the EXC term from the Theorem 3.5 follows. In this case we chose a strong assumption that all t_i (see Theorem 2.1) are equal. This case models a situation where the solver has candidates that take approximately the same amount of time (e.g., a weak/novice solver that only has very general methods to apply in problem solving that equally take a lot of time, or an expert solver that has a set a specific methods with comparable execution time). Note that in this case we state both upper and lower bound with the same assumptions (Theorem 4.4 and Theorem 4.5, respectively).

Theorem 4.4. For each $i = 1, 2, \dots, N$, let $c \leq p_i \leq d$ for some $c, d \in (0, 1)$ and $t_i = T$ for some $T > 0$. Then, the term EXC in Theorem 3.5 can be upper bounded as

$$EXC \leq Td \frac{p_k - p_{k+n}}{1 - p_k} \frac{(1 - c)^k}{c^2} (A - B(1 - c)^{n-1})$$

where

$$A = 1 + kc \left[1 - \frac{1 - p_k}{1 - c} \frac{c}{d} \left(\frac{1 - d}{1 - c} \right)^{k-1} \right],$$

$$B = 1 - c + c(n + k) \left(1 - \frac{c}{d} \right).$$

Proof. First of all, observe that with $t_i = T$ for all $i = 1, 2, \dots, N$ the order of candidates from *Theorem 2.1*, which is given by $(\frac{p_i}{t_i})_{i=1}^N$, depends only on p_i . Thus, by (2.1) we have $p_1 \geq p_2 \geq \dots \geq p_N$, and particularly $p_k \geq p_{k+n}$.

Let q_1 , q_2 , and q_3 be expressions from *Theorem 3.5*. With $p_{k+n} - p_k \leq 0$ and $t_i = T$ for each $i = 1, 2, \dots, N$ we have for q_1

$$q_1 = kT(p_{k+n} - p_k)Q_{k-1},$$

and because $-Q_m \leq -(1-d)^m$, we can write

$$q_1 \leq -kT(p_k - p_{k+n})(1-d)^{k-1}$$

which we transform into

$$q_1 \leq -Td \frac{p_k - p_{k+n}}{1 - p_k} \frac{(1-c)^k}{c^2} \cdot k \frac{1-p_k}{d} \frac{c^2}{1-c} \left(\frac{1-d}{1-c} \right)^{k-1}. \quad (4.6)$$

For q_2 we have

$$\begin{aligned} q_2 &= T \frac{p_k - p_{k+n}}{1 - p_k} \sum_{l=k+1}^{k+n-1} Q_{l-1} l p_l \leq \\ &\leq T \frac{p_k - p_{k+n}}{1 - p_k} d \sum_{l=k+1}^{k+n-1} (1-c)^{l-1} l = \\ &= T d \frac{p_k - p_{k+n}}{1 - p_k} (1-c)^{k-1} \left(\sum_{l=1}^{n-1} l(1-c)^l + k \sum_{l=1}^{n-1} (1-c)^l \right). \end{aligned}$$

By *Lemma 4.1* we can use similar equations to (4.2) and (4.3), and we can further write

$$q_2 \leq T d \frac{p_k - p_{k+n}}{1 - p_k} \frac{(1-c)^k}{c^2} \cdot (1 + kc - (1-c)^{n-1} (1 + c(n+k-1))). \quad (4.7)$$

For q_3 we have

$$\begin{aligned} q_3 &= (k+n)T \frac{p_k - p_{k+n}}{1 - p_k} Q_{k+n-1} \leq \\ &\leq (k+n)T \frac{p_k - p_{k+n}}{1 - p_k} (1-c)^{k+n-1} \end{aligned}$$

which we transform into

$$q_3 \leq T d \frac{p_k - p_{k+n}}{1 - p_k} \frac{(1-c)^k}{c^2} \cdot \frac{c^2}{d} (k+n)(1-c)^{n-1}. \quad (4.8)$$

Finally, by putting (4.6), (4.7), and (4.8) together the theorem follows. \square

Theorem 4.5. For each $i = 1, 2, \dots, N$, let $c \leq p_i \leq d$ for some $c, d \in (0, 1)$ and $t_i = T$ for some $T > 0$. Then, the term *EXC* from the *Theorem 3.5* can be lower bounded as

$$EXC \geq T c \frac{p_k - p_{k+n}}{1 - p_k} \frac{(1-d)^k}{d^2} (A - B(1-d)^{n-1})$$

where

$$A = 1 + kd \left[1 - \frac{1-p_k}{1-d} \frac{d}{c} \frac{e^{-S_{k-1}}}{(1-d)^{k-1}} \right],$$

$$B = 1 - d + d(n+k) \left(1 - \frac{d}{c} \right).$$

Proof. By the same reason as in the beginning of the proof of the *Theorem 4.4* we have $p_k \geq p_{k+n}$. Let q_1 , q_2 , and q_3 be expressions from *Theorem 3.5*. With $p_{k+n} - p_k \leq 0$ and $t_i = T$ for each $i = 1, 2, \dots, N$ we have for q_1

$$q_1 = kT(p_{k+n} - p_k)Q_{k-1},$$

and by *Lemma 3.1* we can write

$$q_1 \geq -kT(p_k - p_{k+n})e^{-S_{k-1}}$$

which we transform into

$$q_1 \geq -Tc \frac{p_k - p_{k+n}}{1-p_k} \frac{(1-d)^k}{d^2} \cdot k \frac{1-p_k}{1-d} \frac{d^2}{c} \frac{e^{-S_{k-1}}}{(1-d)^{k-1}}.$$

The rest of the proof is analogous to the proof of the *Theorem 4.4*. □

Finally, we examine a case in which the probabilities p_i (see *Theorem 2.1*) are all equal. This case models, for example, a situation where the solver has only general problem solving methods, and he is unable to distinguish which one is the most suitable.

Theorem 4.6. *For each $i = 1, 2, \dots, N$, let $p_i = p$ for some $p \in (0, 1)$. Then, the term EXC from the *Theorem 3.5* equals to*

$$EXC = (t_{k+n} - t_k)(1-p)^{k-1} (1 + (1-p) - (1-p)^n).$$

Proof. Let q_1 , q_2 , and q_3 be expressions from *Theorem 3.5*. Then,

$$\begin{aligned} q_1 &= Q_{k-1}(t_{k+n} - t_k) = (t_{k+n} - t_k)(1-p)^{k-1}, \\ q_2 &= \sum_{l=k+1}^{k+n-1} Q_{l-1}p(t_{k+n} - t_k) = \\ &= p(t_{k+n} - t_k)(1-p)^{k-1} \sum_{l=1}^{n-1} (1-p)^l = \\ &= p(t_{k+n} - t_k)(1-p)^k \frac{1 - (1-p)^{n-1}}{p}, \\ q_3 &= 0. \end{aligned}$$

By summing the expressions for q_1, q_2, q_3 we get the final result. □

5 The case with multiple values of t_k

In real life problem solving a particular solution candidate (e.g., a method) could have been used to solve multiple similar problems each time consuming a different amount of time. Therefore, when the solver is considering a potential solution candidate, it has one cumulative probability of success (e.g., based on the past experience and the strength of similarity/relatedness with the current problem model), but it can have multiple application times because of this possible application to the similar problems in the past.

Theorem 5.1. *Let s_k be a solution candidate which we in the past applied n_k times, and let $t_{k,j}$ be the execution time of the j^{th} application. Denote the mean execution time of the solution candidate s_k with Et_k :*

$$Et_k = \frac{t_{k,1} + \dots + t_{k,n_k}}{n_k}.$$

If one continues to select subsequent candidates on the basis of maximum p_k/Et_k , then the expected time before solving the problem will be minimal (provided the problem can be solved by one of our candidates).

Proof. By using the linearity of the expected value, the proof is identical to the proof of the Theorem 2.1. □

References

- [1] E. Fink. Automatic evaluation and selection of problem-solving methods: Theory and experiments. *Journal of Experimental & Theoretical Artificial Intelligence*, 16(2):73–105, 2004.
- [2] E. A. Hansen and S. Zilberstein. Monitoring the progress of anytime problem-solving. In *Proceedings of the National Conference on Artificial Intelligence*, pages 1229–1234, 1996.
- [3] M. Hutter. A theory of universal artificial intelligence based on algorithmic complexity. *arXiv preprint cs/0004001*, 2000.
- [4] M. Hutter. *Universal artificial intelligence: Sequential decisions based on algorithmic probability*. Springer Science & Business Media, 2005.
- [5] M. Klamkin and D. Newman. Extensions of the weierstrass product inequalities. *Mathematics Magazine*, pages 137–141, 1970.
- [6] L. A. Levin. Universal sequential search problems. *Problemy Peredachi Informatsii*, 9(3):115–116, 1973.
- [7] G. Polya. How to solve it. *Princeton University*, 1957.
- [8] S. J. Russell and D. Subramanian. Provably bounded-optimal agents. *Journal of Artificial Intelligence Research*, pages 575–609, 1995.
- [9] J. Schmidhuber. Optimal ordered problem solver. *Machine Learning*, 54(3):211–254, 2004.
- [10] R. Solomonoff. Algorithmic probability: Theory and applications. In *Information Theory and Statistical Learning*. Springer, 2009.

- [11] R. Solomonoff. Algorithmic probability—Its discovery—Its properties and application to strong AI. *Randomness Through Computation: Some Answers, More Questions*, 2011.
- [12] R. J. Solomonoff. Optimum sequential search. *Memorandum, Oxbridge Research, Cambridge, Mass*, 1984.
- [13] R. J. Solomonoff. The application of algorithmic probability to problems in artificial intelligence. In *UAI*, pages 473–494, 1985.
- [14] S. Wu. Some results on extending and sharpening the weierstrass product inequalities. *Journal of mathematical analysis and applications*, 308(2):689–702, 2005.